UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/683,383	12/20/2001	Jens Leenstra	DE920000098US1	9708

877 7590 08/24/2004

IBM CORPORATION, T.J. WATSON RESEARCH CENTER
P.O. BOX 218
YORKTOWN HEIGHTS, NY 10598

EXAMINER	
O BRIEN, BARRY J	
ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 08/24/2004

6

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/683,383	LEENSTRA ET AL.	
	Examiner	Art Unit	
	Barry J. O'Brien	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12/20/01, 2/19/02 and 7/17/03*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 February 2002* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☒ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *5*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-10 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed on record in the file: Drawings as received on 2/19/02, Priority Document as received on 2/19/02, and IDS as received on 7/17/03.

Priority

3. Acknowledgment is made of applicant's claim for foreign priority based on an application filed in the European Patent Office on 12/23/00. It is noted, however, that applicant has not filed a certified copy of the 00128489.2 application as required by 35 U.S.C. 119(b).

Drawings

4. Figures 1-6, 9 and 11 should be designated by a legend such as --Prior Art-- because only that which is old is illustrated (see p.5-6 of the Specification, wherein the descriptions of the drawings identify them as depicting prior art content). See MPEP § 608.02(g). Corrected drawings in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. The replacement sheet(s) should be labeled "Replacement Sheet" in the page header (as per 37 CFR 1.121(d)) so as not to obstruct any portion of the drawing figures. If the examiner does not accept the changes, the applicant will be notified and

informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Specification

5. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

6. The applicant is requested to review the specification and update the status of all co-pending applications made mention of, replacing attorney docket numbers with current U.S. application or patent numbers when appropriate.

7. The specification contains numerous grammatical and English errors, and appears to have been a literal translation of a foreign document. Please review the specification and correct any such errors.

8. The abstract of the disclosure is objected to because of the following informalities:

a. The abstract contains numerous grammatical and English errors, and appears to have been a literal translation of a foreign document. Please review the abstract and correct any such errors. Listed below are some exemplary issues.

b. The abstract recites the phrase, "a high frequency outprocessor". It is unclear what an "outprocessor" is, and it is not defined in the specification. Please correct the language of the abstract to more clearly provide a summary of the invention as claimed.

c. The abstract refers to both "no_dependency" and "no dependency" as seemingly being the same thing. Please correct the language of the abstract to be consistent and only use one phrase to represent this.

d. The abstract talks about the purported merits of the invention. The abstract should not refer to purported merits or speculative applications of the invention and should not compare the invention with the prior art. Please correct the language of the abstract.

Correction is required. See MPEP § 608.01(b).

Claim Objections

9. Claims 2-6 are objected to because of the following informalities:

a. A series of singular dependent claims is permissible in which a dependent claim refers to a preceding claim which, in turn, refers to another preceding claim. A claim that depends from a dependent claim should not be separated by any claim that does not also depend from said dependent claim. It should be kept in mind that a dependent claim may refer to any preceding independent claim. In general, applicant's sequence will not be changed. See MPEP § 608.01(n). See claims 1-5, where claim 5 improperly depends on claim 2, while claims 3-4 depend on claim 1.

b. Claim 2 recites the limitation "dependency-signal" on its last line. There is no need for a dash in this limitation, and its presence makes the claim language unclear. Please correct the claim language to read, "dependency signal".

c. Claim 3 recites the limitation, "no-dependency-signal" on its third line. There is no need for a dash between dependency and signal in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "no-dependency signal".

d. Claim 4 recites the limitation, "to a mapping-table-based renaming scheme comprising the step of:" on its second line. However, the claim lists multiple steps. Please correct the claim language to read, "to a mapping-table-based renaming scheme comprising the steps of:".

e. Claim 4 recites the limitation, "committed-status-flag" on its sixth line. There is no need for a dash between status and flag in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "committed-status flag".

f. Claim 4 recites the limitation, "dependencyfor" on its last line. Please correct the claim language to read, "dependency for".

g. Claim 5 recites the limitation, "to a mapping-table-based renaming scheme comprising the step of:" on its second line. However, the claim lists multiple steps. Please correct the claim language to read, "to a mapping-table-based renaming scheme comprising the steps of:".

h. Claim 5 recites the limitation, "committed-status-flag" on its sixth line. There is no need for a dash between status and flag in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "committed-status flag".

i. Claim 6 recites the limitation, "no-dependency-signal" on its eighth line. There is no need for a dash between dependency and signal in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "no-dependency signal".

j. Claim 8 recites the limitation, "valid-bits" on its second line. There is no need for a dash in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "valid bits".

k. Claim 10 recites the limitation, "no-dependency-signal" on its ninth line. There is no need for a dash between dependency and signal in this claim, and its presence makes the claim language unclear. Please correct the claim language to read, "no-dependency signal".

Appropriate correction is required.

Claim Rejections - 35 USC § 112

10. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

11. Claims 6-10 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

12. Claim 6 recites the limitations, "A first computer readable code", "a second computer readable code", and "a third computer readable code". However, the functions that are described in the claims are enabled in the specification as being performed in hardware, not as instructions (computer readable code). For example, the "generating a no-dependency signal" limitation is actually performed using a series of AND and OR gates, as described in dependent claims 7-9. Claim 10 has the same issues as claim 6. Dependent claims 7-9 contain all of the limitations of their parent claim, and thus are rejected for the same reasons as claim 6.

13. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

14. Claims 1-9 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

15. Claim 1 recites the limitation, "A method for operating an out-of-order processor in which a rename process is comprised of the pipeline an instruction stream is processed with". It is unclear how a "rename process" can comprise a "pipeline", as a pipeline in this context is a physical entity wherein a process is comprised of instructions. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed. Dependent claims 2-5 contain all the limitations of their parent claims, and thus are rejected for the same reasons as claim 1.

16. Claim 1 recites the limitation, "a logic target address of one or more instructions stored in a temporary buffer associated with a pipeline process downstream" on its fifth and sixth lines. It is unclear whether a "logic target address" or "one or more instructions" are being stored in the

temporary buffer downstream. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed. Dependent claims 2-5 contain all the limitations of their parent claims, and thus are rejected for the same reasons as claim 1.

17. Claim 2 recites the limitation, "in case of a match;" on its fourth line. It is unclear what this limitation has to do with the rest of the claim language, as it appears to just be a dangling limitation. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed. Dependent claim 5 contains all the limitations of its parent claim, and thus is rejected for the same reasons as claim 2.

18. Claim 3 recites the limitation, "evaluating "valid" of non-architected target registers stored in a storage associated with speculatively calculated instruction result data into the no-dependency-signal generation". It is unclear how it is possible to evaluate registers "into" a signal generation. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed.

19. Claim 3 recites the limitation, "non-architected target registers" on its second line. Its is unclear what these non-architected target registers are, as their converse, architected target registers, are not defined in the claims. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed.

20. Claim 4 recites the limitation, "in case of a match;" on its eighth line. It is unclear what this limitation has to do with the rest of the claim language, as it appears to just be a dangling limitation. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed.

21. Claim 5 recites the limitation, "in case of a match;" on its eighth line. It is unclear what this limitation has to do with the rest of the claim language, as it appears to just be a dangling limitation. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed.

22. Claim 6 recites the limitation, "post-connected OR gate" on its last line. It is unclear if the "post-connected" part of the OR gate refers to it being connected after (post) another element, or to a "post", or something else altogether, as it is not described sufficiently in the specification. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed. See also similar limitations requiring correction in dependent claims 7 and 9. Dependent claims 7-9 contain all of the limitations of their parent claims, and thus are rejected for the same reasons as claim 6.

23. Claim 8 recites the limitation "the target register "valid bits" signal" in its second line. There is insufficient antecedent basis for this limitation in the claim.

24. Claim 9 recites the limitation, "a logic for ANDing a selected" on its fifth and sixth lines. It is unclear what a "selected" is, and how it could be involved in an AND operation. Please correct the claim language to more clearly define the metes and bounds of the invention as claimed.

Claim Rejections - 35 USC § 103

25. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

26. Claims 1-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carnevale et al., U.S. Patent No. 5,471,626, in further view of Garg et al., U.S. Patent No. 5,974,526.

27. Regarding claim 1, Carnevale has taught a method for operating an out-of-order (see Carnevale, Col.6. lines 28-33) processor in which a process is comprised of the pipeline an instruction stream is processed with (see Carnevale, Figs.2-3 and Col.4 lines 18-24, 47-51), the method comprising the steps of:

a. For detection of a dependency determining for each current instruction involved in a process that a logic target address of one or more instructions stored in a temporary buffer associated with a pipeline process downstream of the current instruction is not the same as a logic source address of said current instruction (see Carnevale, Col.6 lines 5-22),

b. Generating a no-dependency-signal associated with said current instruction (see Carnevale, Col.6 lines 23-28). Here, although an explicit "no-dependency" signal has not been taught, there is inherently a signal representing the presence or absence of a dependency between registers that is used to determine whether or not a stage can be bypassed (see Carnevale, Col.6 lines 5-28).

c. Forwarding said signal for exploiting said signal in order to control the processing of said current instruction in order to bypass a portion of the pipeline (see Carnevale, Col.6 lines 28-39). Here, if no dependency exists between a current instruction and one downstream in the pipeline, the instruction (also, "control

word", see Carnevale, Col.2 lines 31-44 and Col.9 lines 19-25) is bypassed around the unnecessary pipeline stage (see Carnevale, Col.6 lines 28-39).

28. Carnevale has not explicitly taught wherein the process is a rename process.

29. However, Garg has taught using a mapping table-based renaming process when instructions are executed out-of-order so that performance reducing storage conflicts (see Garg, Col.1 line 62 – Col.2 lines 15) can be handled without degrading performance (see Garg, Col.3 lines 11-27 and Col.4 lines 16-40). Because Carnevale has taught the out-of-order execution of instructions (see Carnevale, Col.6 lines 28-33) and the detection of storage hazards (see Carnevale, Col.6 lines 5-28), one of ordinary skill in the art would have found it obvious to modify the processor of Carnevale to use register renaming in order to eliminate storage conflicts, thereby improving processor performance.

30. Regarding claim 2, Carnevale in view of Garg has taught the method according to claim 1, in which the step of generating a no-dependency signal comprises the steps of:

a. Comparing a plurality of logic target register addresses and the logic source register address of the current instruction, in case of a match (see Carnevale, Col.6 lines 5-22),

b. Generating a dependency-signal for the respective source register (see Carnevale, Col.6 lines 23-28). Here, although an explicit "dependency" signal has not been taught, there is inherently a signal representing the presence or absence of a dependency between registers that is used to determine whether or not a stage can be bypassed (see Carnevale, Col.6 lines 5-28).

31. Regarding claim 3, Carnevale in view of Garg has taught the method according to claim 1, further comprising the step of evaluating "valid" of non-architected target registers stored in a storage associated with speculatively calculated instruction result data into the no-dependency-signal generation (see Carnevale, Col.6 lines 5-28). Here, all non-architected pipeline registers (i.e. not in the register file) contain valid bits (see Carnevale, Fig.4) that identify if the contents of said pipeline register is valid (see Carnevale, Col.5 lines 22-45). Thus, the dependency detection evaluates the source register of the current instruction and register values in "valid" non-architected pipeline registers, outputting the results as an indication of "dependency", or "no dependency" (see Carnevale, Col.6 22-28).

32. Regarding claim 4, Carnevale in view of Garg has taught the method according to claim 1, further comprising the step of applying the method to a mapping-table-based renaming scheme comprising the step of:

a. Addressing a mapping table entry with a logical source register address of said current instruction thus determining the mapped physical target register address (see Garg, Col.6 lines 48-63). Here, the TAL is addressed using a logical source address, and outputs "tags" corresponding to physical addresses where the register values are currently located (see Garg, Col.12 lines 30-58).

b. Reading a committed-status-flag in said entry (see Garg, Col.13 lines 50-56),

c. Comparing the logic target register address and the logic source register address of the current instruction (see Carnevale, Col.6 lines 5-28), and in case of a match,

d. Generating a dependency for the respective source register (see Carnevale, Col.6 lines 5-28).

33. Regarding claim 5, Carnevale in view of Garg has taught the method according to claim 2, further comprising the sep of applying the method to a mapping-table-based renaming scheme comprising the step of:

a. Addressing a mapping table entry with a logical source register address of said current instruction thus determining the mapped physical target register address (see Garg, Col.6 lines 48-63). Here, the TAL is addressed using a logical source address, and outputs "tags" corresponding to physical addresses where the register values are currently located (see Garg, Col.12 lines 30-58).

b. Reading a committed-status-flag in said entry (see Garg, Col.13 lines 50-56),

c. Comparing the logic target register address and the logic source register address of the current instruction (see Carnevale, Col.6 lines 5-28), and in case of a match,

d. Generating a dependency-signal for the respective source register (see Carnevale, Col.6 lines 5-28).

34. Regarding claim 6, Carnevale has taught a processing system having means for executing a readable machine language (see Carnevale, Col.3 lines 60-63), said readable machine language comprises:

a. A first computer readable code for, the detection of a dependency, determining for each current instruction involved in a process that a logic target address of one or more instruction stored in a temporary buffer associated with a pipeline process downstream of the current instruction is not the same as a logic source address of said current instruction (see Carnevale, Col.6 lines 5-22),

b. A second computer readable code for generating a no-dependency-signal associated with said current instruction (see Carnevale, Col.6 lines 23-28). Here, although an explicit "no-dependency" signal has not been taught, there is inherently a signal representing the presence or absence of a dependency between registers that is used to determine whether or not a stage can be bypassed (see Carnevale, Col.6 lines 5-28).

c. A third computer readable code for forwarding said signal for exploiting said signal in order to control the processing of said current instruction in order to bypass a portion of the pipeline (see Carnevale, Col.6 lines 28-39). Here, if no dependency exists between a current instruction and one downstream in the pipeline, the instruction (also, "control word", see Carnevale, Col.2 lines 31-44 and Col.9 lines 19-25) is bypassed around the unnecessary pipeline stage (see Carnevale, Col.6 lines 28-39).

35. Carnevale has not explicitly taught wherein the process is a rename process.

36. However, Garg has taught using a mapping table-based renaming process when instructions are executed out-of-order so that performance reducing storage conflicts (see Garg, Col.1 line 62 – Col.2 lines 15) can be handled without degrading performance (see Garg, Col.3 lines 11-27 and Col.4 lines 16-40). Because Carnevale has taught the out-of-order execution of instructions (see Carnevale, Col.6 lines 28-33) and the detection of storage hazards (see Carnevale, Col.6 lines 5-28), one of ordinary skill in the art would have found it obvious to modify the processor of Carnevale to use register renaming in order to eliminate storage conflicts, thereby improving processor performance.

37. Regarding claim 7, Carnevale in view of Garg has taught the processing system according to claim 6, in which in case of a content-addressable memory (CAM)-based renaming scheme (see Garg, Col.8 lines 38-65 and Col.12 lines 42-58) the first computer readable code for determining the dependency of a current instruction comprises a compare logic in which all instructions to be checked for dependency are involved and a post-connected OR gate (see Garg, 206 of Fig.2 and 700 of Fig.7). Here, each generic comparison block (see Garg, 204 of Fig.2) performs three separate comparisons on a current and prior instructions operands (see Garg, Fig.7), and the generic comparison block outputs that a dependency exists if any of the three individual comparisons are true (see Garg, Col.10 line 56 – Col.11 line 46). Further, the mapping-table based renaming system used by Garg (see Garg, Col.8 lines 38-65 and Col.12 lines 42-58) is also inherently content-addressable, as it is indexed into using contents of its registers (see Garg, Col.12 lines 42-58).

38. Regarding claim 8, Carnevale in view of Garg has taught the processing system according to claim 7, further comprising a plurality of AND gates (see Garg, 808 of Fig.8) the input of which comprises the target register "valid-bits" signal (see Garg, 512/514 of Fig.5 and Col.10 lines 56-61, as well as Carnevale, Fig.4 and Col.5 lines 22-45) and a respective compare logic output signal (see Garg, Figs. 2, 7 and 8). Here, there is a plurality of AND gates (see Garg, 808 of Fig.8) because the circuit of Fig.8 is duplicated three times in comparators 702, 704 and 706, as well as many times in the Data Dependency Checker (see Garg, Fig.2 and Col.10 line 56 – Col.11 line 46).

39. Regarding claim 9, Carnevale in view of Garg has taught the processing system according to claim 6, in which the case of mapping-table-based renaming scheme (see Garg,

Col.8 lines 38-65 and Col.12 lines 42-58) each mapping table entry comprises an additional instruction-committed flag (see Garg, "Done Flag", Col.6 lines 48-63 and Col.13 lines 51-56), and the first computer readable code for determining the dependency of a current instruction comprises a logic for ANDing (see Garg, 808 of Fig.8) a selected in which all instructions to be checked for dependency are involved and a post-connected OR gate (see Garg, 206 of Fig.2 and 700 of Fig.7). Here, each generic comparison block (see Garg, 204 of Fig.2) performs three separate comparisons on a current and prior instructions operands (see Garg, Fig.7), and the generic comparison block outputs that a dependency exists if any of the three individual comparisons are true (see Garg, Col.10 line 56 – Col.11 line 46).

40. Regarding claim 10, Carnevale has taught a computer system having an out-of-order processing system (see Carnevale, Col.6. lines 28-33), said computer system executes a readable machine language (see Carnevale, Col.3 lines 60-63), said readable machine language comprises:

a. A first computer readable code for, the detection of a dependency, determining for each current instruction involved in a process that logic target address of one or more instruction stored in a temporary buffer associated with a pipeline process downstream of the current instruction is not the same as a logic source address of said current instruction (see Carnevale, Col.6 lines 5-22),

b. A second computer readable code for generating a no-dependency-signal associated with said current instruction (see Carnevale, Col.6 lines 23-28). Here, although an explicit "no-dependency" signal has not been taught, there is inherently a signal representing the presence or absence of a dependency between

registers that is used to determine whether or not a stage can be bypassed (see Carnevale, Col.6 lines 5-28).

c. A third computer readable code for forwarding said signal for exploiting said signal in order to control the processing of said current instruction in order to bypass a portion of the pipeline (see Carnevale, Col.6 lines 28-39). Here, if no dependency exists between a current instruction and one downstream in the pipeline, the instruction (also, "control word", see Carnevale, Col.2 lines 31-44 and Col.9 lines 19-25) is bypassed around the unnecessary pipeline stage (see Carnevale, Col.6 lines 28-39).

41. Carnevale has not explicitly taught wherein the process is a rename process.

42. However, Garg has taught using a mapping table-based renaming process when instructions are executed out-of-order so that performance reducing storage conflicts (see Garg, Col.1 line 62 – Col.2 lines 15) can be handled without degrading performance (see Garg, Col.3 lines 11-27 and Col.4 lines 16-40). Because Carnevale has taught the out-of-order execution of instructions (see Carnevale, Col.6 lines 28-33) and the detection of storage hazards (see Carnevale, Col.6 lines 5-28), one of ordinary skill in the art would have found it obvious to modify the processor of Carnevale to use register renaming in order to eliminate storage conflicts, thereby improving processor performance.

Conclusion

43. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the

patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

44. Norden et al., U.S. PGPUG No. 2002/0199085, has taught a variable length pipeline that can bypass certain unnecessary pipeline stages.

45. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barry J. O'Brien whose telephone number is (703) 305-5864. The examiner can normally be reached on Mon.-Fri. 6:30am-4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

46. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Barry J. O'Brien
Examiner
Art Unit 2183

BJO
8/20/2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100